Using RSSI to Form Path in an Indoor Space

Muztaba Fuad
Dept. of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
fuadmo@wssu.edu

Debzani Deb
Dept. of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
debd@wssu.edu

Brixx-John G. Panlaqui
Dept. of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
brixxpanlaqui@gmail.com

Charles F. Mickle
Dept. of Computer Science
Winston-Salem State University
Winston-Salem, NC, USA
cmickle118@rams.wssu.edu

***Abstract*—Generating paths of a mobile device in indoor space by sensing its Bluetooth RSSI value is challenging but has real-world applications. Although Bluetooth RSSI suffers from different factors that limit its usability, this research shows that it can still be used to detect mobility and, over a duration of time, can be used to form paths. This poster presents algorithms that can create a path of a moving mobile device by sensing its RSSI values over time and then presents early results of the algorithm's effectiveness while tracking health practitioners' movement within a community care clinic setting.**

Keywords—mobility, path forming, Bluetooth, received signal strength indicator.

I. Introduction

While there are numerous causes of waste in the US healthcare system, some are associated with inefficiency. There have been multiple proposed solutions to address inefficiency, one of which is clinic layout optimization [1]. Such optimization depends on how furniture and instruments are placed in the clinic and how clinicians move through and between different clinic rooms. Traditionally, such optimization research involves manual monitoring by human proctors, which is inaccurate, unproductive, and subjective. If mobility patterns in an indoor space can be determined automatically, such optimization can be improved significantly.

Mobile crowdsensing is a powerful but cheap technology for the pervasive sensing of valuable data that provides solutions to various real-world problems. From a community clinic's perspective, opportunistic crowdsensing data from the clinician can be leveraged to allow more productive clinic layout optimization. Data that such mobile sensing apps can sense include the practitioner's movement within the clinic and their contextual information, such as location, body position, and device analytics. By combining such contextualized movement data within the clinic, appropriate big-data analytics and visualization can extract intelligence and improve this instance of human-centric service delivery by addressing several pressing needs. These needs may include incorporating different clinic layouts to improve patient experience, finding the most efficient way to utilize resources to improve patient contact time, making the practitioner productive in providing patient care, and improving their learning experience.

Traditionally, Bluetooth Low Energy beacons are used for similar indoor localization problems [2]-[3]. Although beacons are cheap and easy to install, they lack the computational smartness and power needed to localize and contextualize the data. This poster presents an approach that combines stationary and mobile devices within an indoor space to detect motion by using a proximity-based approach as utilized in COVID-19 contact tracing. More specifically, this research utilizes basic Bluetooth RSSI values sensed by one or more stationary devices to form a path of a mobile device.

II. Related Work

Mobile crowdsensing is a broad research area, and a plethora of work [4]-[5] was performed in different aspects of this domain. Indoor localization [6] is one of the most important topics addressed by mobile crowdsensing and is recognized as a challenging research area. There are currently several techniques [6] for indoor localization, and extensive research [4]-[7] has been conducted in this area. Although these techniques have some strengths and shortcomings, this research has different goals and approaches. The application domain requires easy application of this technology without sophisticated hardware or the need for precious accuracy or efficient energy usage. Since mobility detection is the goal, the approach does not need high precision. Additionally, the use of stationary devices connected to power gives this approach the flexibility needed to use the Bluetooth proximity detection [8] technique in a useful way. With the incorporation of computationally intensive and power-savvy stationary mobile devices instead of specialized beacons with less computation power, this research presents an approach to indoor mobility detection without the need for any specialized hardware. Although there is a plethora of related work [4]-[7] in this research area, the proposed research is different from others in the following two ways: 1) it explicitly uses Bluetooth proximity detection, as used by COVID-19 exposure notification apps and 2) it does not utilize any specialized hardware or software and focuses on the easy application of this approach in real-world scenarios.

III. Path Forming

Assume that we have *n* stationary devices $S_1, S_2, S_3....S_n$ placed in *n* clinic rooms with their corresponding proximity. Proximity is defined by a RSSI value within a specific range, indicating that a particular device is close. Let us also assume that if the mobile device *D* has moved from proximity 1 to proximity 2 between time t_i and t_j, each stationary device detects the mobile device's RSSI value in both instances. We denote that as P_{st}, where *s* denotes the stationary device and *t* denotes the time. Between a starting time (t_{start}) and an ending time (t_{end}), the collected data by the stationary devices looks like the following:

t_{start}	P_{1start}	P_{2start}	P_{3start}	....	P_{nstart}
….					
….					
t_i	P_{1i}	P_{2i}	P_{3i}	....	P_{ni}
t_j	P_{1j}	P_{2j}	P_{3j}	....	P_{nj}
….					
….					
t_{end}	P_{1end}	P_{2end}	P_{3end}	....	P_{nend}

Once the above data is collected and aggregated by the stationary devices on the edge, the following algorithm creates vectors (with a tail (from) to a head (towards)) that indicate the mobile device's movement after sorting the data by time:

1. The source of the Path (D_{sr}) is the stationary device that has the largest RSSI value at t_{start}.
2. The destination of the Path (D_{ds}) is the stationary device that has the largest RSSI value at t_{end}.
3. For every t_i
   a. get the i^{th} row values and designate that as R
   b. get the j^{th} $(i+1)$ row values and designate that as S
   c. for every RSSI value in R
      i. if $P_{si} < P_{sj}$: This indicates that the device is moving away from the current location (stationary device s) towards the destination (ds), then create a vector $D_s \rightarrow D_{ds}$.
      ii. if $P_{si} > P_{sj}$: This indicates that the device is moving closer to the source (D_{sr}), then create a vector $D_s \rightarrow D_{sr}$.

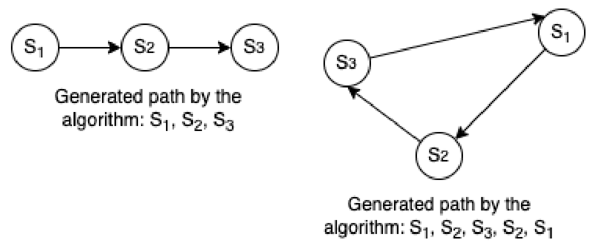
After the above algorithm determines the source and the destination and goes through each timestamped data to create all the vectors, the following algorithm merges all the vectors into a path:

1. Assign D_{sr} in the beginning of the path.
2. For every vector
   a. If the tail is equal to D_{sr} and the head is equal to D_{ds}:
      - This is already in the path, so ignore
   b. If the tail is equal to D_{sr}, but the head is not equal to D_{ds}:
      - Append the head of the vector to the path
   c. If the tail is not equal to D_{sr}, but the head is equal to D_{ds}:
      - Append the tail of the vector to the path
   d. If the tail is not equal to D_{sr}, and the head is not equal to D_{ds}:
      - Append the tail of the vector to the path
      - Append the head of the vector to the path
3. Append D_{ds} to the end of the path.

The above algorithm creates a sequence of stationary device proximities (S_a, S_b, ... S_n), denoting a path through which the mobile device has moved between time t_{start} and t_{end}.

IV. Results, Challenges, and the Future

To test the algorithm's accuracy, we manually generated several sets of simulated data with three stationary devices (S_1, S_2, S_3). Fig. 1 (a) shows one such path and the corresponding path generated by the algorithm. The algorithm identified all the other paths with 100% accuracy. In our next set of experiments, physical devices were used to gather actual RSSI data within a physical space where 3 stationary devices were placed in a hall and a volunteer with a mobile device walked between the devices. Fig. 1 (b) shows that mobility and what the algorithm generates based on these inputs. Similarly, the volunteer physically walked other paths, and data were collected and fed to the algorithm, which identified all the paths with 100% accuracy. However, one thing was observed in the physical experimentation, depicted in Fig. 1 (b). Although the volunteer walked from stationary device S_1 to S_3 via S_2 and back to S_1 again, the path created by the algorithm shows the repetition of stationary device S_2 on the way back to S_1, although the volunteer did not go explicitly to that



(a) (b)

Fig. 1. Sample paths.

device. This is because, while the volunteer is moving from S_3 to S_1, S_2's reception of the RSSI value of the mobile device also changed from low to high. Therefore, the algorithm shows a path from S_3 to S_2 because the volunteer was within the proximity of S_2, even if they did not directly approach it. Currently, the algorithm is not able to distinguish between these two situations. Along with that, the following are couple of other challenges that this research is going to investigate in the future:

- The algorithm can work with five stationary devices right now. It needs to be generalized so that it can work with any number of stationary devices.
- How to detect the RSSI values better, given that the environment has lot of noise from other nearby devices.
- All the stationary devices sense the RSSI values of the mobile device asynchronously because of how the app in the stationary devices is scheduled by the operating system. Although this seems to be a benefit, further investigations need to be performed to see the effect of synchronized data collection.

Acknowledgment

This research is supported by NSF fund #2131100 and by UNC Research Opportunities Initiative (ROI) award "Winston-Salem State University Center for Applied Data Science", FY 2020 – 2023.

References

[1] Vahdat, V., Namin, A., Azghandi, R., & Griffin, J. (2019). Improving patient timeliness of care through efficient outpatient clinic layout design using data-driven simulation and optimization. Health Systems, 8(3), 162–183.

[2] Pušnik, Maja, Mitja Galun, and Boštjan Šumak. "Improved Bluetooth low energy sensor detection for indoor localization services." Sensors 20.8 (2020): 2336.

[3] Chen, J., Zhang, Y., & Xue, W. (2018). Unsupervised indoor localization based on smartphone sensors, iBeacon and Wi-Fi. Sensors, 18(5), 1378.

[4] Du, Y. (2020). Collaborative Crowdsensing at the Edge (Doctoral dissertation, Sorbonne Université).

[5] Liu, C., Hua, J., & Julien, C. (2019, March). Scents: Collaborative sensing in proximity iot networks. 2019 IEEE Intl. Conf. on Pervasive Computing and Communications Workshops (pp. 189-195). IEEE.

[6] Zafari, F., Gkelias, A., & Leung, K. K. (2019). A survey of indoor localization systems and technologies. *IEEE Communications Surveys & Tutorials*, *21*(3), 2568-2599.

[7] Chen, J., Zhang, Y., & Xue, W. (2018). Unsupervised indoor localization based on smartphone sensors, iBeacon and Wi-Fi. Sensors, 18(5), 1378.

[8] Montanari, A., Nawaz, S., Mascolo, C., & Sailer, K. (2017, March). A study of bluetooth low energy performance for human proximity detection in the workplace. In 2017 IEEE International Conference on Pervasive Computing and Communications (PerCom) (pp. 90-99). IEEE.